

## 6.1 Typregeln I

Gegeben seien die Klassen aus der Vorlesung:

```
class C11 {
    C12 v;
}

class C12 { ...}
```

Leiten Sie mit den Typregeln aus der Vorlesung für den Block

```
{
    C11 x;
    return x.v;
}
```

den korrekten Typ her.

## 6.2 Typchecker

Programmieren Sie die Funktionen

```
typecheckStmt :: Stmt -> [(String, Type)] -> [Class] -> Stmt
```

für die abstrakte Syntax von Aufgabenblatt 3.

**Beispiel für If:**

```
typecheckStmt :: Stmt -> [(String, Type)] -> [Class] -> Stmt
typecheckStmt (If(be, ifs, Nothing)) symtab cls =
    let
        bexp = typecheckExpr be symtab cls
        ifstmt = typecheckStmt ifs symtab cls
    in
        if ((getTypeFromExpr bexp) == "boolean") then
            TypedStmt(If(bexp, ifstmt, Nothing), getTypeFromStmt ifstmt)
        else
            error "boolean expected"
```

```
getTypeFromExpr :: Expr -> Type
getTypeFromExpr (TypedExpr(_, typ)) = typ
```

```
getTypeFromStmt :: Stmt -> Type
getTypeFromStmt (TypedStmt(_, typ)) = typ
```

**Hinweis:** Für die Expressions müssen Sie eine Funktion

```
typecheckExpr :: Expr -> [(String, Type)] -> [Class] -> Expr
```

programmieren.

### 6.3 Typregeln II (Prüfungsaufgabe 10 Punkte)

Gegeben seien die Klassen aus der Vorlesung:

```

class C11 {
    char m1 () {
        int b;
        C12 x = new C12 ()
        return x.m2(x.v, b);
    }
}

class C12 {
    C13 v;
    char m2(C13 v, int w) { ...}
}

class C13 { ...}

```

Leiten Sie für den Block

```

{
    int b;
    C12 x = new C12 ()
    return x.m2(x.v, b);
}

```

mit den Regeln aus der Vorlesung den Typ `char` her.

**Hinweis:**

⋮

⋮

$$\frac{
 \begin{array}{c}
 [\text{BlockLVarD}] \\
 [\text{BlockLVarD}]
 \end{array}
 }{
 \frac{
 \{b : \text{int}\} \triangleright_{\text{Stmt}} \text{Block}(\text{LVarDecl}(x, CL2); \text{Assign}(\text{LorFVar}(x), \text{New}(CL2)); \text{Return}(\text{MethodCall}(\text{LorFVar}(x), m2, (\text{InstVar}(\text{LorFVar}(x), v), \text{LorFVar}(b)))))) : \text{char}
 }{
 \emptyset \triangleright_{\text{Stmt}} \text{Block}(\text{LVarDecl}(b, \text{int}); \text{LVarDecl}(x, CL2); \text{Assign}(\text{LorFVar}(x), \text{New}(CL2)); \text{Return}(\text{MethodCall}(\text{LorFVar}(x), m2, (\text{InstVar}(\text{LorFVar}(x), v), \text{LorFVar}(b)))))) : \text{char}
 }
 }$$